SOFTWARE & SYSTEMS QUALITY CONFERENCES International | 2007

Title of Paper

Comparison of Approaches to the Test of Control Unit Software

Presenter

Dr. Olaf Maibaum / Deutsches Zentrum für Luft- und Raumfahrt (D)

Instructional Level

O Introductory **X** Intermediate **O** Advanced

Target Group

Test Practitioners and Engineers, Software and Test Managers, Design and Development Engineers

Keywords

- Software in the Loop (SiL)
- Simulation based test
- Test of fault scenarios
- Test automation

Abstract

Introduction

Control units in a safety-critical area must fulfil a lot of requirements concerning the reliability and the robustness. These requirements must be fulfilled in a raising magnitude by the control unit software. To guarantee the compliance of these requirements, the control unit software is tested with external malfunctions, as for example due to noisy sensors, disturbed communication or corrupt actuators. [1]

Besides, the software under test is coupled with a simulation of the dynamic system behaviour. So, it is necessary to execute the tests in a closed control loop. According to the kind of the coupling between software under test and environment we have to distinguish between "hardware in the loop" (HiL), "processor in the loop" (PiL) "software in the loop" (SiL), and “model in the loop” (MiL) tests.

Within the scope of the project SiLEST the SiL test approach is compared to the HiL test approach. The test approach is evaluated with regard to the applicability and recommendations are given for the application of the test process.

This article explains the different test approaches and gives an overview of the SiLEST test execution environment developed for the SiL test. The test environment permits the automatic execution of tests of control unit software without manual interventions. Afterwards we describe our experiences with the new test approach which we made with the testing of an automotive engine management system and an attitude control system of a small satellite. Finally we give in these paper recommendations for the application of the different test approaches.

Test approaches

In a HiL test the software under test is executed directly on a control unit which resembles the later used control unit. The simulation of the dynamic system behaviour is connected by the original hardware interfaces. The big disadvantages of this test approach are that the tests can be executed only at a very late development stage and that a cost-intensive real-time hardware is necessary. With this test method statements about the temporal conformity of the software under test can be met.

Within a PiL test the software to be tested is executed on a control unit which is similarly to the later used control unit. Besides, the simulation of the dynamic system behaviour is connected by alternative hardware interfaces instead of the original ones. To make these alternative interfaces available, changes are necessary, as for example in the hardware access level of the software under test. These changes cause another timing behaviour of the software under test. Without further efforts to the temporal synchronisation of the software to be tested and the simulation of the dynamic system behaviour no statements about the temporal conformity of the software under test are possible. The advantage of this test approach is that it can be applied earlier and more economically in the development than the HiL approach.

With a SiL test the control unit software under test is embedded in the simulation of the dynamic system behaviour. Through this the hardware expenses are omitted for the test of the control unit software. It is a big advantage of the SiL test approach that it is also possible to test models of the control unit software. Through this the functionalities of the control unit software can be already checked very early in the development process. Also the test of partial integrated control unit software is possible. Besides, the integration of the functionalities follows gradually through the replacement of models by software modules, until the control unit software is implemented completely. By the automation of the test execution regression tests can be executed in parallel with the development to reach a high software quality. Statements about the temporal conformity of the control unit software cannot be met with this test approach.

Within a MiL test the software under test is exchanged by a functional model of the software. This approach allows a very early test of the functionalities. The simulation environment for this approach is the same as for the SiL test. This allows a mix of the SiL and MiL test on partly integrated software.

SiLEST test execution environment

To allow the SiL test without manual interventions we developed a test execution environment within the scope of the project SiLEST. This environment is suitable for the execution of software tests with the SiL test approach, and also for other test approaches like MiL, PiL, and HiL. A short introduction to the test approach is given in [2]. The test execution environment can be customised to the different requirements of the project partners from different application areas. Within the scope of SiLEST these are the areas of the automotive engineering and the space technology. For the project partners it is important that different software tools, i.e. commercial standard software as well as open-source tools, can be integrated flexibly into the test execution environment. We used within the project for example Mercury TestDirector™ or Testmaster as tools for the test case management.

The test execution environment automates all activities in the complete test process. These are:

- the access to test suites and test cases from the test management system;
- the access to the simulation components and the software under test which stand under configuration management;
- the initialisation of the simulation and the software to be tested;
- the test execution including the operation of additional test support tools;
- the automatic analysis of the test results;
- the generation of test reports;
- the archiving of the test reports safe about audit.

The test cases are defined in a XML format. A test case contains continuous or discrete input signals and the expected results of the test case beside general information on the test case. The test case format was described in detail in [3].

Besides, the flexibility of the test execution environment is reached by a plug-in technology for the integration of the different tools and the use of a script defining the basic test execution. If modifications in the test execution for other application domains are necessary, these can be made by changes in the script.

Conclusions and overview

In the project SiLEST a test execution environment was developed for the SiL test method. Special attention was laid to the fact that also a faulty behaviour of the sensors and actuators can be simulated. This allows the assurance of the robustness of the control unit software under these faults.

To mention the advantages and disadvantages of the SiL test compared to the HiL test, the same SiL and HiL tests are executed for control unit software and are compared with each other in our presentation. The main focus of the presentation lays on the experiences made with the SiL test of the attitude control software of the BIRD micro satellite [4,5]. The test results are compared with the recorded behaviour of the satellite in space. From these experiences we give recommendations for the application of the SiL tests.

The tests carried out with the defined test execution environment up to now show the applicability early in the development process and allow an accompanying test of the model based development process. Besides, a cost reduction for the test is reached. The reasons are the re-use of test cases and simulation models and the automated execution of the tests. A concluding evaluation of the test approach can be given at the end of the project SiLEST in April 2007.

References

[1] Testing the Robustness of Spacecraft Control Software. O. Maibaum. Ercim News Nr. 65. pp 17-18. April 2006.

[2] Automated Closed-Loop Testing of Embedded Engine Control Software. S. Rebeschieß, T. Liebezeit, U. Bazarsuren. ICS Test 2006.

[3] XML Format for Automated Software-in-the-Loop Tests. T. Liebezeit, H. Schumann, S. Rebeschieß, C. Gühmann, U. Bazarsuren. ICS Test 2006.

[4] Technology Demonstration by the BIRD Mission. Brieß et. al. Erschienen in Digest of the 4th Int. Symposium of the IAA, Berlin April 7 - 11, 2003. pp. 51-54. Wissenschaft & Technik Verlag. 2003.

[5] Lessons learned from the object-oriented design of the Bird attitude control system software. O. Maibaum, Olaf, T. Terzibaschian. In 16th IFAC Symposium on automatic control in aerospace, International Federation of automatic control. Nebylov, Alexander [Hrsg.]. pp 156 – 161. St. Petersburg. 2004.

Biography

Dr. Olaf Maibaum has studied computer science and business economics at the university of Hildesheim finalized with a diploma in 1996.

From 1996 till 1997 he worked as research assistant at the institute of operating systems and network communication at University of Hildesheim. From 1998 to 1999 he was employed as research assistant at the institute of operating systems and distributed systems at the Carl von Ossietzky University of Oldenburg. The research activities from 1996 till 2002 were the calculation of the worst case execution times of processes in embedded real time systems. In 2003 he obtain the PHD in computer science at the university of Oldenburg with the PHD thesis „Bestimmung symbolischer Laufzeiten in eingebetteten Echtzeitsystemen“ („Determination of symbolic runtimes in embedded real time systems“).

Since 2000 Dr. Olaf Maibaum worked at the german aerospace centre (DLR) at Braunschweig. He is involved in the software engineering for onboard space applications. He also performs research actities in software testing of embedded real time systems. He is the project manager of the project SiLEST (Software in the Loop for embedded system testing).

Contact information of Presenter

Dr. Olaf Maibaum
Lilienthalplatz 7
38108 Braunschweig
Deutsches Zentrum für Luft- und Raumfahrt e.V in der Helmholtz-Gemeinschaft
Olaf.Maibaum@dlr.de
Phone: +49(0)531/295-2974
Fax: +49(0)531/295-2767